LA-UR -91-2336

Conf-910794--62

LA-UR--91-2386

DE91 016016

TITLE AN APPLICATION OF NEURAL NETWORKS TO PROCESS AND MATERIALS CONTROL

AUTHOR(S) Jo Ann Howell and Rena Whiteson

SUBMITTED TO 32nd Annual Meeting of the Institute of Nuclear Materials Management, New Orleans, Louisiana, July 28-31, 1991

Los Alamos

Los Alamos National Laboratory
Los Alamos, New Mexico 87545

[illegible]

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

AN APPLICATION OF NEURAL NETWORKS TO PROCESS AND MATERIALS CONTROL*

J. A. Howell
R. Whiteson
Safeguards Systems Group
Los Alamos National Laboratory
Los Alamos, New Mexico 87545

ABSTRACT

Process control consists of two basic elements: a model of the process and knowledge of the desired control algorithm. In some cases the level of the control algorithm is merely supervisory, as in an alarm-reporting or anomaly-detection system. If the model of the process is known, then a set of equations may often be solved explicitly to provide the control algorithm. Otherwise, the model has to be discovered through empirical studies. Neural networks have properties that make them useful in this application. They can learn (make internal models from experience or observations).

The problem of anomaly detection in materials control systems fits well into this general control framework. To successfully model a process with a neural network, a good set of observables must be chosen. These observables must in some sense adequately span the space of representable events, so that a signature metric can be built for normal operation. In this way, a non-normal event, one that does not fit within the signature, can be detected. In this paper, we discuss the issues involved in applying a neural network model to anomaly detection in materials control systems. These issues include data selection and representation, network architecture, prediction of events, the use of simulated data, and software tools.

INTRODUCTION

Modern safeguards systems for nuclear materials handling typically use distributed control with operator control consoles.[1] Such systems may involve several levels of control: data acquisition (for displaying current plant information), supervisory control (operator issuing commands), and continuous control (maintaining a device at a given state automatically). Because of the complexity of the processes and the large and diverse amount of data, efficient automatic algorithms are necessary to interpret the data and ensure secure plant operation.[2]

Data analysis techniques used to ascertain plant conditions must not only recognize normal facility operations, but also be able to respond to non-normal (anomalous) conditions. This requires having a good understanding of the underlying processes. With this understanding we can build a model for anomaly detection that is based on the same hypothesis as that used in computer intrusion detection:[3] exploitation of system vulnerabilities involves non-normal system use. That is, if we can build a model of normal system behavior, then we can detect non-normal or unsecure system behavior.

---

*This work supported by the U. S. Department of Energy, Office of Safeguards and Security.

A necessary feature of an automated safeguards system is the ability to detect an anomalous event, identify the nature of the event, and recommend a corrective action. In this paper we propose and discuss a new technique for the first step: detection. This technique, based on neural networks, has been applied to certain aspects of the anomaly detection problem in computer security. [4] We utilize this technique to enhance the real-time materials control aspects of safeguards systems.

Although there are many safeguards systems in place, the anomaly detection problem has received only moderate attention.[5] Suggested approaches include statistical analysis, pattern recognition, and rule-based methods.[2,5] Our approach, using neural networks,[6] is based on the ability of a network to model complex, nonlinear, real-time processes. It is in the form of a predictive method in which we predict a future state of the system based on present and past states after one or more system parameters (valves, pumps, etc.) are changed. This prediction is the focus of the work under way. We hope to determine the requirements for a near-real-time anomaly detection system that would integrate materials control, materials accounting, process control, and security system data.

THE MATERIALS CONTROL PROBLEM

A nuclear materials safeguards system must be capable of detecting anomalies. The system should be robust enough to detect, assess, and respond to a non-normal situation. Several factors make this task extremely difficult.

1. Nuclear materials facilities tend to be very complex with a large number and variety of instruments. This results in the production and storage of huge amounts of data, making timely human review of all data impossible.
2. There may be more possible normal operating states than could ever occur in the entire existence of the facility.
3. There may be even more non-normal operating states than normal ones. It would be impossible to predict or describe all possible non-normal states.
4. The data from facility instrumentation tends to be noisy and sometimes even incorrect.

Our approach is based on the hypothesis that non-normal transactions or states in a facility would involve unusual patterns of behavior. Anomalies that we would like to be able to detect include:

1. Theft or diversion of material from the system;
2. Protracted diversion of very small amounts of material from the system;
3. Unauthorized access to measurement records, measurement control records, or transaction records; and
4. Instrument failures.

All of these activities would involve anomalous behavior of some part of the system that is being monitored: tank volumes, tank levels, flow rates, valves, pump and steam jet states, pressure meters, infrared and ultrasonic meters, motion detectors, etc. A good model of the process is necessary to detect these states and locate and describe them.

MODELING AND CONTROL OF PROCESSES BY NEURAL NETWORKS

To model a plant process, we are using neural computing, a method that is an attempt to create a computer model that matches the functionality of the human brain. These models assume that information processing takes place through the interactions of many simple processing units, each sending excitatory and inhibitory signals to other units. In traditional expert systems, knowledge is explicitly stored in the form of rules. The generation of rules is not a well understood process; cloture is difficult to ascertain. Neural networks, on the other hand, generate their own rules by learning from examples. They tend to be robust because knowledge is distributed uniformly around the network, with information being processed in a parallel manner. Another feature is their efficiency; they can utilize large amounts of data in a near-real-time manner. The complexity of the plant being modeled can range from a very simple plant with few devices and instruments to a very complex plant with highly nonlinear relationships between the variables. Once the network has been designed and tested on a small test problem, it can be expanded easily to model an entire plant.

There are two steps in the operation of a network: learning and recall. Learning is the process of modifying the connections in response to "training" examples presented to the net. These examples are a set of observables that provide a signature metric for normal activity. This set must be chosen very carefully so that in the multivariate space of parameters describing the plant, the set adequately spans the variable space. A network is given several example inputs together with their desired outputs. During this iterative process, the connections in the network are changing and adapting. When the connections are no longer changing (learning has ceased), the network is trained. Such learning is called "supervised learning." During the testing phase, samples of a test data set are presented to the network, and an output is produced. Comparing this output with the desired output gives an estimate of the error to be expected when presenting unknown data to the trained network.

The trained network has, during this iterative procedure, formed a model indicating the relationship of the outputs as a function of the inputs. It can model highly nonlinear processes, given the correct internal architecture. By examining the trained network, one can determine the relative importance of the individual input values. Thus, one can iteratively refine the data selection process, eliminating those input parameters that have little or no effect.

In our work, we build a simple model of plant operation with the state of various instruments for the past and present as inputs and the state of these instruments for the future as outputs.

DATA REPRESENTATION

The data we are using as a training data set are based on data from a chemical processing plant. The process monitoring system is a set of instruments and sensors installed on plant equipment that transmit process data to a computer for processing and storage. Data are collected from pneumatic and electronic instruments and digital controllers. Readings from the tanks represent volumes, and readings from the valves, pumps, and steam jets represent on/off or open/closed status. Of interest at this time are vessel volumes and valve pump, and steam jet states. Because these data are extremely noisy and occasionally erroneous, we use simulated data while in the development phase.

For simplicity, tanks are called tank1, tank2, ..., tankN. Valves, pumps, and steam jets are labelled valve1, valve2, valve3, ..., valveN. We simulated two different transfer scenarios.

1. Material is transferred between tank1 and tank2.
2. Material is transferred from tank1 or tank2 out of the system.

Scenario 1 represents normal material transfers, while scenario 2 represents a non-normal transfer in which material is lost from the system.

Transfer scenarios are generated by a simulation that randomly selects the source tank, randomly selects the amount of material to be transferred, and randomly chooses the time of day of the transfer, with the constraint that there is exactly one transfer per 12 hour period.

From the transfer scenarios we generate simulated data records for a cell in the plant. This cell has two tanks with valves, pumps, and steam jets that control material flow. The simulation arbitrarily sets the initial volume of tank1 to 300 units of material and tank2 to 400 units. Records are generated for every 4-minute period; each record consists of a time stamp, the volumes of the tanks, and the states of the valves. A steady state is maintained (all valves closed, no changes in volume) between transfers. A transfer scenario spans 3 to 8 records depending on the amount of material transferred. The following sequence of events occurs during the transaction: valves associated with the source tank open, the source tank volume begins to decrease, valves associated with the destination tank open, the destination tank volume begins to increase, source tank volume stabilizes, source tank valves close, destination tank volume stabilizes, destination tank valves close. Because some material often remains in transit after valves have closed, there is some discrepancy between the amount of material that leaves the source tank and the amount that enters the destination tank. The simulation generates data for a three month period, producing 33 120 records.

THE NEURAL NETWORK MODEL

Neural networks have demonstrated an impressive ability to deal with the modeling of the type of problems discussed here. At present, the most popular net for function approximation is a feed-forward back-propagation network.[4] This net is composed of input and output layers and one or more hidden layers of neurons (Fig. 1).

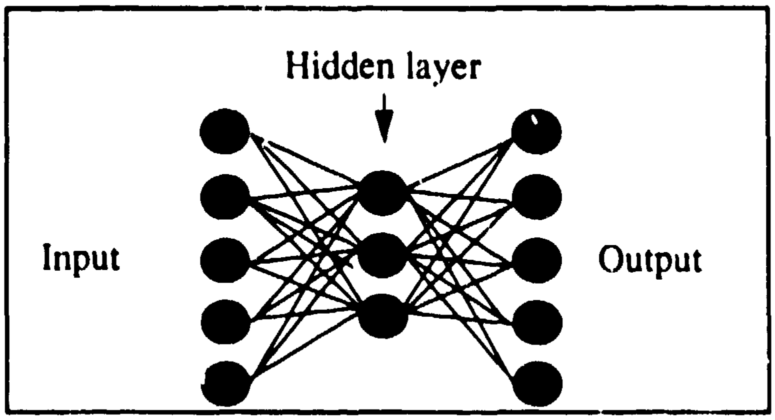

Figure 1. Neural network.

All of the nodes in a given layer are processed simultaneously, with information flowing in a single direction. The output y_i of the ith neuron is given by

$$y_i = sig(\sum_j W_{ij} y_j + \Theta_i)$$

where y_j is the output of the jth neuron in a layer immediately to the left of the layer in which the ith neuron is located. The sigmoid function *sig* (called a transfer function) is defined by

$$sig(x) = \frac{1}{2}[1 + \tanh(x)]$$

The form of this function is chosen to mimic, in a rough sense, biological neurons. The weights W_{ij} and thresholds Θ_i are determined by least-mean-squares minimization. Define a cost function

$$E = \frac{1}{2}\sum_{p=1}^{M} [f(x_p) - \phi(x_p)]^2$$

where x_p is an input training vector, $f(x_p)$ is the training output for the input vector x_p and $\phi(x_p)$ is the network output for the training input x_p. The summation is over all training points. M is the number of times that any training point is shown to the net. For convenience, we have assumed a single output, although there can be multiple outputs. The learning algorithm is simply the numerical technique for the minimization of E. Common minimization methods, for instance, are gradient descent, conjugate gradient, and Newton's method.[7] Figure 2 shows several inputs, a single neuron in a hidden layer, and a single output.

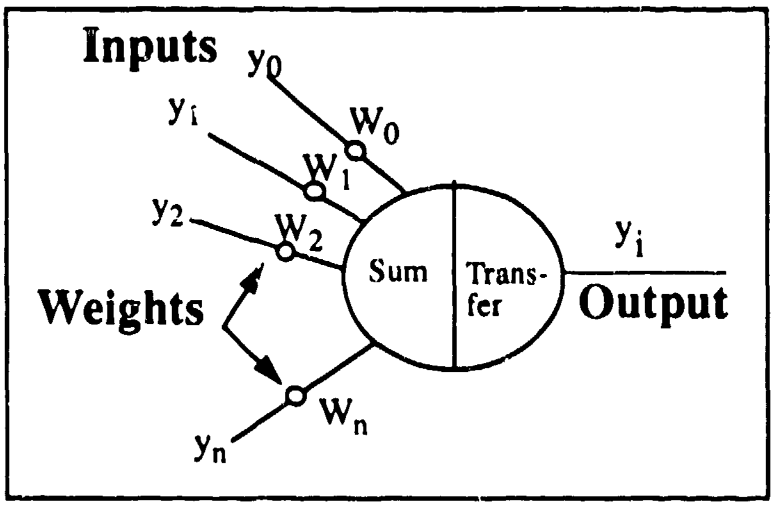

Figure 2. A single neuron and its inputs and output.

Back-propagation networks have had some impressive successes. This class of networks has been able to out-perform nearly all the traditional methods in the accuracy of time-series prediction.[8,9] The objective is to determine the value of the time series at some future time, given a number of past and present values. Because this is analogous to our problem of predicting the state of a plant, it is an ideal method to use for our experiments.

SIMULATION AND RESULTS

To train the network, we created 24 instances of scenario 1 and generated data records from those scenarios. To test the network, we created 12 instances of material transfers, 75% of which transfers were normal (scenario 1) and 25% were non-normal (scenario 2). The non-normal transactions are randomly interspersed with the normal transactions. We then generated data records from these scenarios.

From the simulated records we extracted selected fields and created new data records that would be the input to the network. There were two factors in our choice of fields: we wanted the network to predict current tank volumes and our scenarios involved only tank1 and tank2. The fields we selected were the volumes in tank1 and tank2 and the states of the remote valves (2), pumps (2), and steam jets (3) associated with those tanks.

The new records that we created have 27 fields:

7 antepenultimate valve states,

7 penultimate valve states,

7 current valve states,

2 antepenultimate tank volumes,

2 penultimate tank volumes, and

2 current tank volumes.

Using a commercial software package NeuralWorks Plus, we configured the network with 25 input nodes. These correspond to fields 1-25 of our data records. There is one hidden layer with 16 nodes. The output layer has two nodes, corresponding to the current volumes of the two tanks. These are the values the network is predicting. (See Figure 3.)

We trained the network on the records from 12 days of plant operation, including 24 transfers of material. With a total of 4800 records, the network was trained on 50 000 inputs from shuffled input data. The network was then tested on records from 6 days of plant operation, including 12 transfers of material. The data set for testing consists of 2200 records that are shown to the network sequentially. For each record, the network predicts the current volume in the two tanks.

After testing, an error checking routine passes through the output from the network. When one or more tank volume changes, it indicates the beginning of material transfer. When the system returns to a steady state, if the predicted tank volumes differ from the volumes reported by the instruments by more than some predetermined tolerance, the transfer is flagged as anomalous and a loss of material is indicated. A running total of material discrepancy is maintained.

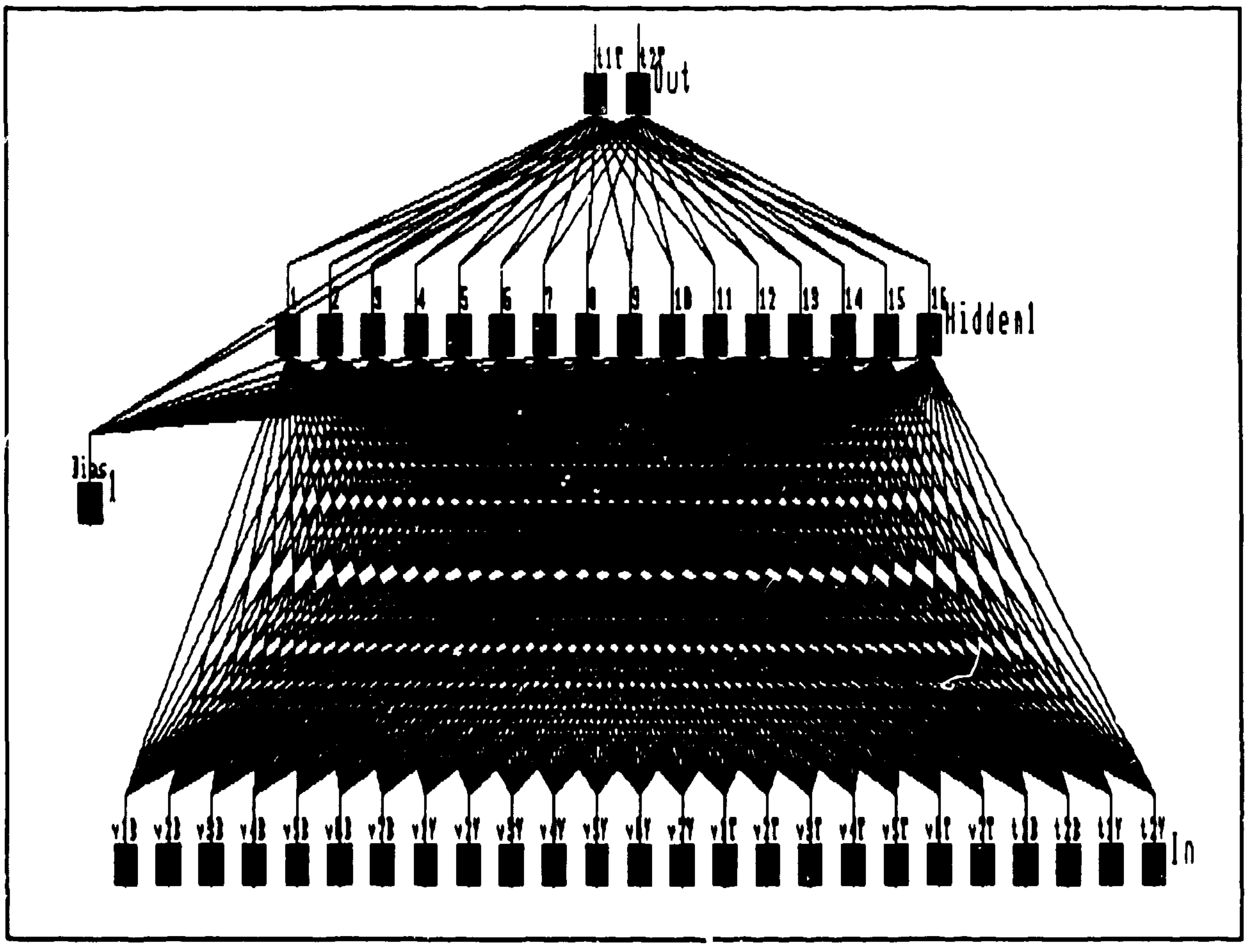

Figure 3. Neural network for anomaly detection.

Table I shows the transfer scenarios, the actual amount lost from the source tank, the actual amount gained by the destination tank, the predicted loss from the source tank, the predicted gain by the destination tank, and a discrepancy. Each time the system returns to a steady state, a disparity is computed. This is the difference between the sum of the actual tank volumes and the sum of the predicted tank volumes. After each transaction, a discrepancy is computed. This is the difference between the current disparity and the disparity before the transaction. If the network's prediction of tank volumes diverges from the reported volumes, a loss is indicated.

Note that scenarios 3, 7, and 11 are non-normal and represent a loss of material from the system. In scenario 3, 62 units are taken from tank2 and disappear. The network predicts a loss of 39.36 units from tank2 and a gain of 35.49 units for tank1. At the conclusion of the transaction, both tanks' reported volumes diverge from the predicted volumes; by comparing those amounts, a safeguards security officer would conclude that approximately 56.55 units of material had been diverted. Interestingly, even after an anomalous transaction, normal transactions are recognized as such.

TABLE 1: Scenarios

Scenario Requested	Actual Amount		Predicted Amount		Discrepancy
	Loss	Gain	Loss	Gain	
1. From: Tank1 To: Tank2 Amount: 26.00	27	27	27.61	28.14	0.53
2. From: Tank2 To: Tank1 Amount: 68.00	70	72	72.09	71.40	1.11
3. From: Tank1 To: Outside system Amount: 58.00	62	0	39.36	35.49	56.55*
4. From: Tank2 To: Tank1 Amount: 43.00	46	44	44.91	44.56	2.36
5. From: Tank1 To: Tank2 Amount: 11.00	12	11	12.16	11.44	0.18
6. From: Tank2 To: Tank1 Amount: 9.00	10	9	9.23	9.73	1.50
7. From: Tank1 To: Outside system Amount: 55.00	58	0	37.44	32.85	53.40*
8. From: Tank2 To: Tank1 Amount: 43.00	46	44	44.30	45.91	3.61
9. From: Tank1 To: Tank2 Amount: 77.00	81	80	82 38	82.66	1.28
10. From: Tank2 To: Tank1 Amount: 49.00	53	52	55.14	52.71	1.43
11. From: Tank1 To: Outside system Amount: 13.00	14	0	8.93	7.94	13.01*
12. From: Tank2 To: Tank1 Amount: 43.00	46	44	44.05	47.14	5.09

*Anomalous transaction.

Figure 4 shows graphs for both the predicted and actual volumes in Tank1 and Tank2 in the first four scenarios. The graphs are scaled for clarity. The first graph shows the predicted volumes of Tank1, while the second shows the actual volumes as reported by the control system. The next two graphs show the same information for Tank2.

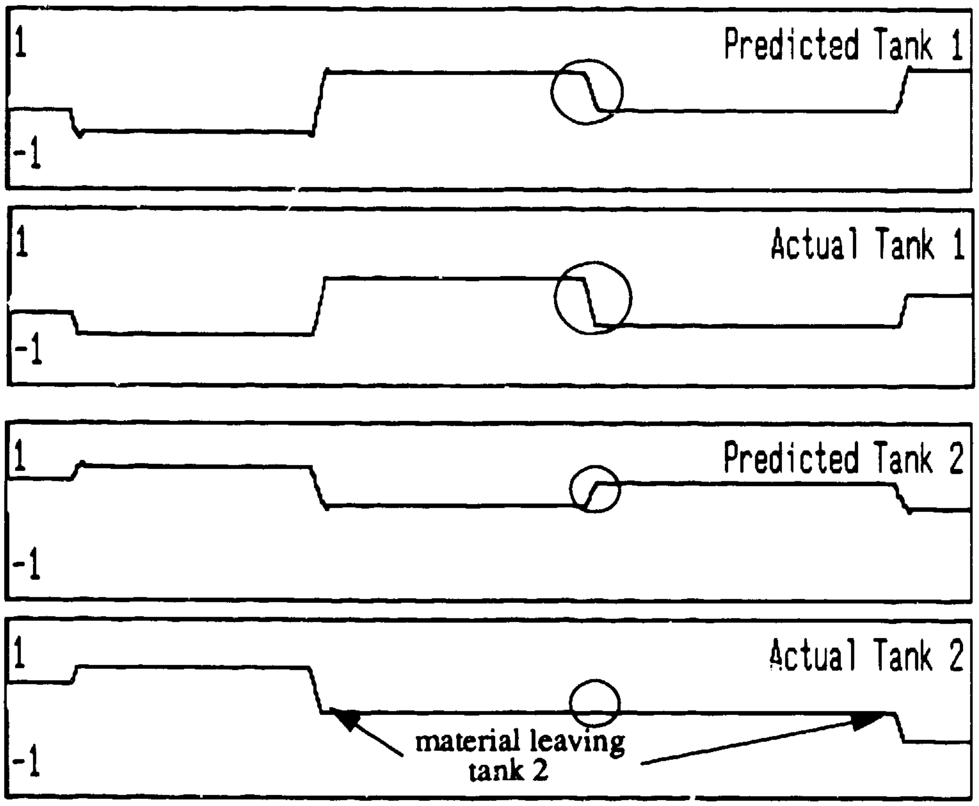

Figure 4. Results of tests.

Scenario 3, which is anomalous, is circled. It represents an instance in which material was transferred from a tank to outside the system. The network was trained to predict that when material leaves a tank, it should appear in another. When presented with a test case in which material is transferred from a tank to *outside* the system, it results in an error, the amount of which denotes what is missing. This error appears in a data file for analysis.

CONCLUSIONS AND FUTURE WORK

Our experiment with neural networks on simulated data from a process monitoring system indicates that the neural network approach may offer an efficient and reliable algorithm for material control and accounting. If a trained network represents a good model of normal plant operation, it can be a reliable tool for recognizing non-normal activity.

We have shown that neural networks can be an effective tool in anomaly detection. Using simulated data, our test network has successfully detected loss of material from a closed system and approximated the amount of material lost. Before we attempt to enhance the capabilities of the network, we need to optimize it.[10] To do so we must determine

1. How far the network must look back,
2. Appropriate initial weights,
3. Best learning rule,
4. Best transfer function, and
5. The optimal number of nodes in the hidden layer.

Once the network has been optimized, we can begin to expand its functions. Foremost, the network must be able to monitor a large and complex system with many tanks and with different kinds of inputs. Besides tank volumes and valve states, instruments report pressures, flow rates, tank levels, etc. The network must be expanded to include input data from more plant instruments and specialty sensors than it currently reads.

At present, the network trains on instances of only one kind of normal scenario. For a network to be robust it would have to be trained on data from a large number and type of normal transfer scenarios. Some of these scenarios would represent material entering or leaving the system. Our hope is that such a trained network would successfully recognize any non-normal activities that it encounters.

Another important goal is to develop the network so that it will function with a high degree of accuracy even when given noisy input. This is essential if the network is to function in the real world.

In summary, neural networks offer an exciting alternative methodology for materials control and accounting. Anomaly detection implemented via neural networks can be used to detect material loss. With the ability to analyze data in parallel, they offer an efficient methodology for processing the large amounts of data generated by complex facilities. They could be a tool for realizing real-time integrated safeguards.

REFERENCES

1. M. H. Ehinger, N. R. Zack, E. A. Hakkila, and F. Franssen, "Use of Process Monitoring For Verifying Facility Design For Large Scale Preprocessing Plant," Los Alamos National Laboratory report LA-12149-MS (1991).

2. A. Zardecki, R. Whiteson, and A. Coulter, "Pattern Recognition Methods for Anomaly Detection," Los Alamos National Laboratory, Safeguards Systems Group report N-4/91-420 (April 9, 1991).

3. D. E. Denning, "An Intrusion-Detection Model," *IEEE Transactions on Software Engineering*, SE-13(2), 222-232 (Feruary, 1987).

4. K. L. Fox, R. R. Henning, J. H. Reed, and R. P. Simonian, "A Neural Network Approach Towards Intrusion Detection," *Nucl. Mater. Manage.* XVII, 125-134 (1989).

5. J. T. Markin, R. B. Strittmatter, H. S. Vaccaro, and W. J. Whitty, "Concepts for Advanced MC&A Systems," *Nucl. Mate. Manage.* **XV**(2), 48-54 (January 1987).

6. D. E. Rumelhart and J. L. McClelland, *Parallel Distributed Processing* (MIT Press, Cambridge, Massachusetts, 1989).

7. W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes* (Cambridge University Press, Cambridge, 1986).

8. A. Lapedes and R. Farber, "How Neural Nets Work," in *Neural Information Processing Systems*, D. Z. Anderson, Ed. (American Institute of Physics, New York, 1988), pp. 442-456.

9. A. Lapedes and R. Farber, "Nonlinear Signal Processing Using Neural Networks and System Modelling Prediction," Los Alamos National Laboratory document LA-UR-87-2662 (August 1987).

10. E. B. Baum and D. Haussler, "What Size Net Gives Valid Generalization?" *Neural Computation* **1**(1), 151-160 (Spring 1989).